

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-059231

(43)Date of publication of application : 25.02.2000

(51)Int.Cl.

H03M 7/30  
G10L 19/00

(21)Application number : 10-225609

(71)Applicant : HITACHI LTD

(22)Date of filing : 10.08.1998

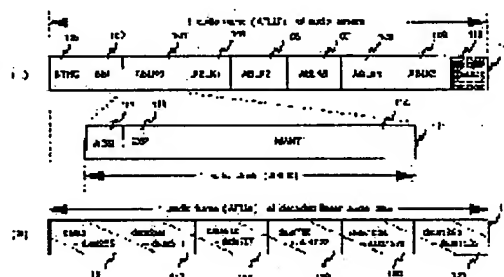
(72)Inventor : OBATA SHINICHI  
FUJII YUKIO

## (54) METHOD FOR COMPENSATING COMPRESSED AUDIO ERROR AND DATA STREAM REPRODUCING DEVICE

## (57)Abstract:

PROBLEM TO BE SOLVED: To connect smoothly recovered parts from an error compensation frame by storing amplitude coefficient data of a final block of one frame and comparing the data with an amplitude coefficient of a first block of a frame succeeding to a frame where errors take place to set a required amplitude.

SOLUTION: A (fn+1)-th frame of a compressed audio stream is received in an audio stream AFLM(fn+1) input step. An error detection code EDC 110 or the like is used to detect presence of an error and the result is stored in a way that the frame numbers of erroneous frames are recognized. Whether or not a (fn)-th frame has been an erroneous frame is confirmed in the AFLM (fn) error confirmation step. In the case that the frame has no error, an amplitude is reproduced based on data EXP 113 relating to amplitude data from information included in the AFLM(fn) in the AFLM(fn) amplitude data reproduction step. Then a sample value RECON-SAMP s reproduced based on the reproduced amplitude and a MANT 114 included in the AFLM(fn). A final block ABLK(5) in the (fn)-th frame is stored.



## LEGAL STATUS

[Date of request for examination]

22.07.2002

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-59231

(P2000-59231A)

(43) 公開日 平成12年2月25日 (2000.2.25)

(51) Int. Cl. <sup>7</sup>	識別記号	F I	テーマコード* (参考)
H 0 3 M 7/30		H 0 3 M 7/30	Z
G 1 0 L 19/00		G 1 0 L 9/18	G

審査請求 未請求 請求項の数10 O L (全 13 頁)

(21) 出願番号 特願平10-225609

(22) 出願日 平成10年8月10日 (1998.8.10)

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 小畑 信一

神奈川県横浜市戸塚区吉田町292番地 株

式会社日立製作所マルチメディアシステム

開発本部内

(72) 発明者 藤井 由紀夫

神奈川県横浜市戸塚区吉田町292番地 株

式会社日立製作所マルチメディアシステム

開発本部内

(74) 代理人 100068504

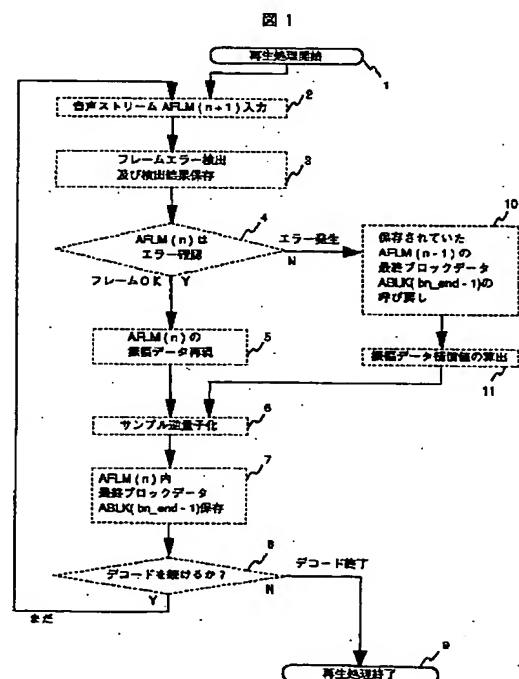
弁理士 小川 勝男

(54) 【発明の名称】 圧縮音声エラー補償方法およびデータストリーム再生装置

(57) 【要約】

【課題】 圧縮音声ストリームに対するエラー補償処理を、より細やかに且つ簡単な方法で行い、エラー発生時の聴感の向上を行う。

【解決手段】 圧縮音声の1フレームの最後のブロックの振幅係数データを記憶しておき、エラーが発生したフレームの次のフレームの最初のブロックの振幅係数と少なくともつながるように、エラーフレームの各ブロックに対する振幅係数を設定する。信号レベルに影響を与えるデータ以外は前フレームの同ブロックを繰り返しとする。



## 【特許請求の範囲】

【請求項1】圧縮の基本単位である圧縮音声フレームの先頭を示す同期ワードと、上記圧縮音声フレームがエラーを含んでいるかどうかを検出するための誤り検出ワードと、更なる細かいデータ区分である複数のオーディオデータブロックとで構成される上記圧縮音声フレーム、或いは上記圧縮音声フレームが複数連続した圧縮音声ストリームを入力し、上記圧縮音声フレーム内に含まれる同期ワードによって上記圧縮音声フレームの先頭を判別し、上記1フレーム分が判別された圧縮音声フレームについて上記圧縮音声フレーム内に含まれる上記誤り検出ワードにより、当該圧縮音声フレームがエラーを含んでいるかどうかを判別し、上記圧縮オーディオフレーム内に含まれる上記データブロックの中に含まれる振幅値と正規化サンプル値を使ってサンプル値を逆量子化して、帯域合成処理をした結果の音声信号を出力する圧縮音声再生方法において、

フレームエラーの履歴を保持し、(fn)番目圧縮音声フレームにエラーが検出されている場合に(fnは整数)、(fn-1)番目圧縮音声フレームの(bn\_end-1)番目ブロック(bn\_endは1フレーム内の総ブロック数)のデータと、(fn+1)番目圧縮音声フレームの(0)番目ブロックのデータから(fn)番目圧縮音声フレームの(0)番目ブロック～(bn\_end-1)番目ブロックのデータに対するエラー補償データを生成することを特徴とした圧縮音声エラー補償方法。

【請求項2】圧縮の基本単位である圧縮音声フレームの先頭を示す同期ワードと、上記圧縮音声フレームがエラーを含んでいるかどうかを検出するための誤り検出ワードと、更なる細かいデータ区分である複数のオーディオデータブロックとで構成される上記圧縮音声フレーム、或いは上記圧縮音声フレームが複数連続した圧縮音声ストリームを入力し、上記圧縮音声フレーム内に含まれる同期ワードによって上記圧縮音声フレームの先頭を判別し、上記1フレーム分が判別された圧縮音声フレームについて上記圧縮音声フレーム内に含まれる上記誤り検出ワードにより、当該圧縮音声フレームがエラーを含んでいるかどうかを判別し、上記圧縮オーディオフレーム内に含まれる上記データブロックの中に含まれる振幅値と正規化サンプル値を使ってサンプル値を逆量子化して、帯域合成処理をした結果の音声信号を出力する圧縮音声再生方法において、

フレームエラーの履歴を保持し、(fn)番目圧縮音声フレームにエラーが検出されている場合に(fnは整数)、(fn-1)番目圧縮音声フレームの(bn\_end-1)番目ブロック(bn\_endは1フレーム内の総ブロック数)の振幅値データと、(fn+1)番目圧縮音声フレームの(0)番目ブロックの振幅値データから(fn)番目圧縮音声フレームの(0)番目ブロック～(bn\_end-1)番目ブロックの振幅値データに対するエラー補

償データを生成することを特徴とした圧縮音声エラー補償方法。

【請求項3】圧縮の基本単位である圧縮音声フレームの先頭を示す同期ワードと、上記圧縮音声フレームがエラーを含んでいるかどうかを検出するための誤り検出ワードと、更なる細かいデータ区分である複数のオーディオデータブロックとで構成される上記圧縮音声フレーム、或いは上記圧縮音声フレームが複数連続した圧縮音声ストリームを入力し、上記圧縮音声フレーム内に含まれる同期ワードによって上記圧縮音声フレームの先頭を判別し、上記1フレーム分が判別された圧縮音声フレームについて上記圧縮音声フレーム内に含まれる上記誤り検出ワードにより、当該圧縮音声フレームがエラーを含んでいるかどうかを判別し、上記圧縮オーディオフレーム内に含まれる上記データブロックの中に含まれる振幅値と正規化サンプル値を使ってサンプル値を逆量子化して、帯域合成処理をした結果の音声信号を出力する圧縮音声再生方法において、

フレームエラーの履歴と(bn\_end-1)番目ブロック(bn\_endは1フレーム内の総ブロック数)の振幅値データを保持し、(fn)番目圧縮音声フレームにエラーが検出されている場合に(fnは整数)、(fn-1)番目圧縮音声フレームの最終である(bn\_end-1)番目ブロックの振幅値データと、(fn+1)番目圧縮音声フレームの先頭である(0)番目ブロックの振幅値データから(fn)番目圧縮音声フレームの(0)番目ブロック～(bn\_end-1)番目ブロックの振幅値データに対するエラー補償データを生成することを特徴とした圧縮音声エラー補償方法。

【請求項4】圧縮の基本単位である圧縮音声フレームの先頭を示す同期ワードと、上記圧縮音声フレームがエラーを含んでいるかどうかを検出するための誤り検出ワードと、更なる細かいデータ区分である複数のオーディオデータブロックとで構成される上記圧縮音声フレーム、或いは上記圧縮音声フレームが複数連続した圧縮音声ストリームを入力し、上記圧縮音声フレーム内に含まれる同期ワードによって上記圧縮音声フレームの先頭を判別し、上記1フレーム分が判別された圧縮音声フレームについて上記圧縮音声フレーム内に含まれる上記誤り検出ワードにより、当該圧縮音声フレームがエラーを含んでいるかどうかを判別し、上記圧縮オーディオフレーム内に含まれる上記データブロックの中に含まれる振幅値と正規化サンプル値を使ってサンプル値を逆量子化して、帯域合成処理をした結果の音声信号を出力する圧縮音声再生方法において、

フレームエラーの履歴と(bn\_end-1)番目ブロック(bn\_endは1フレーム内の総ブロック数)の振幅値データを保持し、(fn)番目圧縮音声フレームにエラーが検出されている場合に(fnは整数)、(fn-1)番目圧縮音声フレームの最終である(bn\_end-1)番目

ブロックの振幅値データ $EXP(f_n - 1, 5, \text{band})$ と、 $(f_n + 1)$ 番目圧縮音声フレームの先頭である $(0)$ 番目ブロックの振幅値データ $EXP(f_n + 1, 0, \text{band})$ とで表現される式

【数1】 $EXP(f_n, b_n, \text{band}) = EXP(f_n - 1, 5, \text{band}) * (6 - b_n) / 7 + EXP(f_n + 1, 0, \text{band}) * (b_n + 1) / 7$  に従って $(f_n)$ 番目圧縮音声フレームの $(0)$ 番目ブロック～ $(b_{n\_end} - 1)$ 番目ブロックの振幅値データに対するエラー補償データを生成することを特徴とした圧縮音声エラー補償方法。

【請求項5】圧縮の基本単位である圧縮音声フレームの先頭を示す同期ワードと、上記圧縮音声フレームがエラーを含んでいるかどうかを検出するための誤り検出ワードと、更なる細かいデータ区分である複数のオーディオデータブロックとで構成される上記圧縮音声フレーム、或いは上記圧縮音声フレームが複数連続した圧縮音声ストリームを入力し、上記圧縮音声フレーム内に含まれる同期ワードによって上記圧縮音声フレームの先頭を判別し、上記1フレーム分が判別された圧縮音声フレームについて上記圧縮音声フレーム内に含まれる上記誤り検出ワードにより、当該圧縮音声フレームがエラーを含んでいるかどうかを判別し、上記圧縮オーディオフレーム内に含まれる上記データブロックの中に含まれる振幅値と正規化サンプル値を使ってサンプル値を逆量子化して、帯域合成処理をした結果の音声信号を出力する圧縮音声再生方法において、フレームエラーの履歴と $(b_{n\_end} - 1)$ 番目ブロック $(b_{n\_end}$ は1フレーム内の総ブロック数)の振幅値データを保持し、 $(f_n)$ 番目圧縮音声フレームにエラーが検出されている場合に $(f_n)$ は整数)、 $(f_n - 1)$ 番目圧縮音声フレームの最終である $(b_{n\_end} - 1)$ 番目ブロックの振幅値データ $EXP(f_n - 1, 5, \text{band})$ と、 $(f_n + 1)$ 番目圧縮音声フレームの先頭である $(0)$ 番目ブロックの振幅値データ $EXP(f_n + 1, 0, \text{band})$ とで表現される式

【数2】 $EXP(f_n, b_n, \text{band}) = EXP(f_n - 1, 5, \text{band}) * 1 / (2^{(1 + b_n)}) + EXP(f_n + 1, 0, \text{band}) * 1 / (2^{(6 - b_n)})$

に従って $(f_n)$ 番目圧縮音声フレームの $(0)$ 番目ブロック～ $(b_{n\_end} - 1)$ 番目ブロックの振幅値データに対するエラー補償データを生成することを特徴とした圧縮音声エラー補償方法。

【請求項6】請求項1から5のいずれか1項記載の圧縮音声エラー補償方法において、特に入力される上記圧縮音声フレームは、時間軸上一定の第1の間隔の音声サンプルに対して圧縮エンコードされたもので、上記圧縮音声フレーム内の上記オーディオデータブロックは上記第1の間隔を更に細かく一定な時間軸上一定の第2の間隔で区切った音声サンプルに対して対応するように圧縮データが納められているものであることを特徴とした圧縮

音声エラー補償方法。

【請求項7】請求項1から5のいずれか1項記載の圧縮音声エラー補償方法において、特に帯域合成処理は、ポリフェーズフィルタバンクによるフィルタリングである、またはポリフェーズフィルタバンクによるフィルタリングを含む処理であることを特徴とした圧縮音声エラー補償方法。

【請求項8】請求項1から5のいずれか1項記載の圧縮音声エラー補償方法において、特に帯域合成処理は、IDFT(離散フーリエ逆変換)である、またはIDFT(離散フーリエ逆変換)を含む処理であることを特徴とした圧縮音声エラー補償方法。

【請求項9】請求項1から5のいずれか1項記載の圧縮音声エラー補償方法において、特に帯域合成処理は、IMDCT(離散フーリエ逆変換)である、またはIMDCT(離散フーリエ逆変換)を含む処理であることを特徴とした圧縮音声エラー補償方法。

【請求項10】圧縮の基本単位である圧縮音声フレームの先頭を示す同期ワードと、上記圧縮音声フレームがエラーを含んでいるかどうかを検出するための誤り検出ワードと、更なる細かいデータ区分である複数のオーディオデータブロックとで構成される上記圧縮音声フレーム、或いは上記圧縮音声フレームが複数連続した圧縮音声ストリームを入力し、圧縮音声フレームの同期検出を行い、上記圧縮音声フレームがエラーを含んでいるかどうかを判別して、フレームエラーフラグを出力するフレーム同期および誤り検出手段と、上記フレーム同期および誤り検出手段による誤り検出が済んだ後の圧縮音声フレームデータからストリーム補助情報、ブロック補助情報、データ振幅値コード、正規化サンプルデータコード等の情報を抜き出して保持するフレーム内情報抜き出し手段と、上記フレーム内情報抜き出し手段から振幅値コードを入力してデータ振幅値をデコードする振幅値デコード手段と、振幅値デコード手段から出力されるデコードされた振幅値を入力し、上記フレーム内情報抜き出し手段から正規化サンプル値を入力して、逆量子化処理を行う逆量子化手段と、上記逆量子化手段から出力されるサンプル再構成値を各帯域について入力し、帯域合成して、最終的な時間領域信号に変換して出力する帯域合成手段とで構成される圧縮音声再生装置において、更に上記フレーム同期および誤り検出手段からの上記フレームエラーフラグを入力して保持する誤り履歴保持手段と、上記振幅値デコード手段でデコードされた振幅値のうち、上記圧縮音声フレーム内での最終オーディオデータブロックに対応する振幅値を入力して保持する過去フレーム最終ブロック保持手段と、 $(f_n)$ 番目圧縮音声フレームに対する逆量子化を行うときに $(f_n)$ は整数)、 $(f_n - 1)$ 番目圧縮音声フレームの最終オーディオデータブロックに対応する振幅値を上記過去フレーム最終ブロック保持手段から入力し、また上記振幅値デコー

ド手段から  $(fn + 1)$  番目圧縮音声フレームの先頭オーディオデータブロックに対応する振幅値を入力して、振幅値補償データを生成して出力する補償値生成手段と、上記誤り履歴保持手段から  $(fn)$  番目圧縮音声フレームに対するフレームエラーフラグを入力して、上記フレームエラーフラグがエラーでないことを示す場合は上記振幅値デコード手段が出力する  $(fn)$  番目圧縮音声フレームの各オーディオデータブロックに対応する振幅値を上記フレームエラーフラグがエラーであることを示す場合は上記補償値生成手段が出力する補償値としての各オーディオデータブロックに対応する振幅値を選び、逆量子化手段に送る振幅値切り替え手段とを備えることを特徴としたデータストリーム再生装置。

#### 【発明の詳細な説明】

【発明の属する技術分野】ディジタルオーディオ再生装置、圧縮音声デコード処理装置 (MPEG-1 audio, MPEG-2 audio) 等、特に、圧縮音声データにエラーが生じたときの、エラーデータ補償方法及び装置に関する。

【従来の技術】従来の圧縮音声再生方法においては、圧縮音声ストリーム中にエラーが検出された場合の対処法としてのエラー補償処理は、圧縮の基本単位であるフレームについて、前のデータを繰り返す、或いはエラー部分をミュートするという処理を行っている。この一例としては、「1992年電子情報通信学会秋季大会、B-571、MPEG/Audio 符号化方式における伝送エラー補償法のハードウェアによる評価：北畠 他3、NEC」に発表がされており、近年でもエラー補償法についてはあまり変化がない。

【発明が解決しようとする課題】例えば、ISO/IEC 11172-3 に示された MPEG-1 audio layer2 で  $Fs = 48$  kHz の場合には、圧縮の基本単位である1フレームは1152サンプル分、24 ms、1フレームは更に細かなブロック (384 サンプル分：8 ms) に分かれており、時間軸データから圧縮データ (周波数軸データ) への変換の最短変換長は1ブロックとなっている。誤り検出ワードはこの1フレームを単位につけられており、エラーが発生した場合は、それが圧縮データであるが故に、エラーの影響は大抵の場合1フレーム全体におよび、1部に抑えこむことはできない。この区間を前フレームの繰り返しや、エラー補償処理 (レベル0化) を行ってしまうと、音源データの特性によっては違和感が知覚される。しかし、エラー補償値の算出のために細かな聴覚特性を利用した演算を導入すると、デコード処理ハードウェアの規模やソフト演算処理量の大幅な増大を招いてしまう。

【課題を解決するための手段】圧縮音声の1フレームの最後のブロックの振幅係数データを記憶しておき、エラーが発生したフレームの次のフレームの最初のブロックの振幅係数と比較する。この両端の振幅係数になだらかなつながるようにエラーフレームの各ブロックに対する

振幅係数を設定する。振幅係数などに代表される信号レベルに影響を与えるデータ以外は前フレームの同ブロックを繰り返すとする。

【発明の実施の形態】以下に本発明の実施形態の一例を示す。図1は圧縮音声再生処理方法の処理手順の一例である。この圧縮音声再生処理に入力される圧縮音声ストリームとしては図2の(1)に示したようなものがある。図2の(1)の圧縮音声ストリームは、圧縮音声の圧縮基本単位であるフレーム101を単位に形成されており、フレームの時間長さは32 msでオーディオブロックはフレームの時間長さの1/6である。フレーム101内はフレーム101の先頭を示すSYNC102、ストリーム補助情報であるBSI103、1フレーム時間内を6分割したオーディオブロックABLK(0)104、ABLK(1)105、ABLK(2)106、ABLK(3)107、ABLK(4)108、ABLK(5)109、さらに誤りを検出するための誤り検出コードEDC110とで構成されている。オーディオブロックABLK(bn)内は更にオーディオブロック情報ABI112、振幅値データEXP113、正規化サンプル値MANT114とで構成されている (bnはブロック番号)。この圧縮音声フレーム101を再生処理すると再生後データ115となり、ABLK(0)104に対するデコード部分は第0再生データ区分116に、ABLK(1)105に対するデコード部分は第1再生データ区分117に、ABLK(2)106に対するデコード部分は第2再生データ区分118に、ABLK(3)107に対するデコード部分は第3再生データ区分119に、ABLK(4)108に対するデコード部分は第4再生データ区分120に、ABLK(5)109に対するデコード部分は第5再生データ区分121になる。図1の説明に戻ると、ステップ1の再生処理開始から再生処理が始まり、まずステップ2の音声ストリームAFLM(fn+1)入力で圧縮音声ストリームの  $(fn+1)$  番目フレームを入力する (fnは整数)。ステップ3のフレームエラー検出及び検出結果保存により、誤り検出コードEDC110などを使い誤りの有無を検出し、その結果を何番目が誤りのあるフレームかわかる形で保存する。ステップ4のAFLM(fn)エラー有無確認で (fn) 番目のフレームがエラーだったかどうかを確認し、エラーなしならばステップ5へ、エラーありならばステップ10へ進む。ステップ5のAFLM(fn)の振幅データ再現ではAFLM(fn)内に含まれた情報から振幅データに関連するデータEXP113をもとに振幅値を再現する。ステップ6のサンプル逆量子化では上記EXP113をもとに再現された振幅値とAFLM(fn)内に含まれたMANT114からサンプル値RECON\_SAMPを再現する。EXP113、MANT114、サンプル値RECON\_SAMPの関係の一例としては下記数式3などがあり、(B

ASE)の(EXP)乗にMANTを掛け合わせるという形で表現される。下記数式4は数式3での(BASE)が2の場合の具体例で、(BASE)が少数や負の数の例も考えられる。

【数3】 $\text{RECON\_SAMP}=(\text{BASE})^{(\text{EXP})}*\text{MANT}$

【数4】 $\text{RECON\_SAMP}=2^{(\text{EXP})}*\text{MANT}$

ステップ6に続いてステップ7のフレーム内最終ブロックデータ保存で(f n)番目のフレーム内の最終ブロックABLK(5)が保存される。ステップ8のデコード継続判断でデコードを続けるかどうか判断され、続ける場合にはステップ2から処理を続け、終了する場合にはステップ9の再生処理終了で再生を終了する。ステップ4において(f n)番目のフレームがエラーだった場合はステップ10の補償用データ呼び戻しで、既に過去に通ったステップ7によって保存されている(f n-1)番目フレームの最終ブロックデータABLK(5)を呼び戻す。ステップ11の振幅データ補償値の算出で、(f n-1)番目フレームの最終ブロックデータABLK(5)と(f n+1)番目フレームの先頭ブロックデータABLK(0)とからエラーフレームである(f n)番目のフレーム内の各ブロックデータABLK(b n)を作り出す。この作り出されたブロックデータABLK(b n)をもとにステップ6以降の処理を続ける。ステップ11内を更に詳しく示したのが図2の処理フローで、ステップ10からの処理の続きはステップ12のAFLM(f n-1)内のABLK(5)の振幅データ入力で(f n-1)番目フレームの最終ブロックデータABLK(5)の振幅データEXP(f n+1, 0, band)を入力する(bandは帯域番号)。続いてステップ13のAFLM(f n+1)内のABLK(0)の振幅データ入力で(f n+1)番目フレームの先頭ブロックデータABLK(0)の振幅データEXP(f n+1, 0, band)を入力する。ステップ14のブロック内振幅値補償データ生成で、エラーフレームであるAFLM(f n)内のABLK(b n)内の補償値を各帯域ごとに生成する。ステップ14に続いてはステップ15の全ブロック終了判断で、最終ブロックまで処理が終わったかどうかを判断し、まだ終わっていないならばステップ14の処理を次のブロックABLKに対して行い、全部終わっていれば、ステップ16の振幅値補償データ保存で、所定の場所に振幅値補償データを置く。続いてステップ6へと処理を進め以下、上記の説明と同様である。

【数5】 $\text{EXP}(f n, b n, \text{BAND})=\text{EXP}(f n-1, 5, \text{BAND})*(6-b n)/7+\text{EXP}(f n+1, 0, \text{BAND})*(b n+1)/7$

【数6】 $\text{EXP}(f n, 2, \text{BAND})=\text{EXP}(f n-1, 5, \text{BAND})*4/7+\text{EXP}(f n+1, 0, \text{BAND})*3/7$

ステップ14内での補償値の生成の仕方に関しては、更に具体的に数式5にその一例が示してある。これは振幅データEXP(f n, b n, band)を振幅データE

XP(f n-1, 5, band)と振幅データEXP(f n+1, 0, band)から作り出す式で、AFLM(f n)内のABLK(b n)内の振幅データEXP各帯域ごとにそれぞれならかに隣のフレームの振幅データEXPへとつながるように加重平均が取られている。ここで、例えば具体的に、ブロック2については、数式5は数式6と等価になる。この効果を図で示したのが図5であり、ブロック内のある決まった帯域に対する振幅データEXPに着目して表現してある。従って、例えば圧縮処理時に帯域分割処理で32個の帯域に分けられて、その1帯域に対して1つの振幅データEXPが決められているのなら、図5のような処理が32の各帯域ごとに行われることとなる。図5の(1)に補償値生成前、図5の(2)に補償値生成後を示している。図5の(1)において、(f n-1)番目フレームのABLK(b n)内のある決まった帯域bandの振幅データはEXP(f n-1, 0, band)126、EXP(f n-1, 1, band)127、EXP(f n-1, 2, band)128、EXP(f n-3, 0, band)129、EXP(f n-1, 4, band)130、EXP(f n-1, 5, band)131で、(f n+1)番目フレームのABLK(b n)内のある決まった帯域bandの振幅データはEXP(f n+1, 0, band)132、EXP(f n+1, 1, band)133、EXP(f n+1, 2, band)134、EXP(f n+1, 3, band)135、EXP(f n+1, 4, band)136、EXP(f n+1, 5, band)137である。エラーフレームである(f n)番目フレームの振幅データについては数式3の処理を行った結果、図5(2)に示したように(f n)番目フレームのABLK(b n)内のある決まった帯域bandの振幅データEXP(f n, 0, band)138、EXP(f n, 1, band)139、EXP(f n, 2, band)140、EXP(f n, 0, band)141、EXP(f n, 4, band)142、EXP(f n, 5, band)143は、ならかにEXP(f n-1, 5, band)131やEXP(f n+1, 0, band)132につながるように生成される。このようにして、エラーフレームに対するエラー補償処理が行われる。これにより、エラー区間と有効フレーム区間とのつながりがよりスムーズになる。また扱うデータはエラーの生じているフレームの長さより、かなり短い区間に対するデータで且つ、更にサンプルを再現する前の部分データで済むため、処理規模の大幅な増大とはならない。またサンプルを再現する前の部分データであるため直行逆変換をする前の周波数領域でのエラー補償となり、時間域での滑らかなつながりだけでなく、周波数域でのスペクトル分布の滑らかなつながりが実現できる。ここで従来型のフレーム繰り返しの例である図6、従来型のミュートの例である図7と図5

(2)を比較すれば効果は明らかで、図6の例ではスペクトル変動の状況によってはEXP131からEXP(f<sub>n</sub>, 0, band)146への分布の急変が違和感を生み出しかねない状態であり、図7の例では明らかにスペクトル分布の急変が生じている。もちろん時間域では窓がけ及びオーバーラップ処理によって滑らかにつながることはできるが、そこから更に聴感を向上させるには、波形のみならずスペクトル分布も滑らかにつながる事が有効である。次に、エラーが2フレーム以上連続した場合の、実施例の効果を図8、図9に示す。まず、図8は過去のAFLM(f<sub>n-1</sub>)が有効データフレームであり、AFLM(f<sub>n</sub>)とAFLM(f<sub>n+1</sub>)がエラーフレームであるときの例である。この場合にも、本発明はその威力を発揮する。つまり、エラーフレームであるAFLM(f<sub>n+1</sub>)内の振幅データEXP(f<sub>n+1</sub>, 0, band)150を暫定的に0と解釈すれば、図8の例のように滑らかにミュートさせることができる。また、この減衰のさせかたが緩やかすぎると判断されるならば、数式5の代わりに数式7のような処理で補償データを作ることも可能である。数式5ではブロックがすすむごとに前フレームからの影響部分が1/2のべき乗で減っていくものとなる。振幅データEXP(f<sub>n</sub>, 3, band)の場合の具体例が数式8である。この場合、減衰の仕方が線形ではなく、対数的になるので聴感特性との親和性が高くなる。

【数7】 $EXP(f_n, b_n, BAND) = EXP(f_{n-1}, 5, BAND) * 1 / (2^{(1+b_n)}) + EXP(f_{n+1}, 0, BAND) * 1 / (2^{(6-b_n)})$

【数8】 $EXP(f_n, 3, BAND) = EXP(f_{n-1}, 5, BAND) * 1 / (2^4) + EXP(f_{n+1}, 0, BAND) * 1 / (2^3)$

但し、 $2^{(b_n)}$ は2の(b<sub>n</sub>)乗

また図9では、過去のAFLM(f<sub>n-1</sub>)とAFLM(f<sub>n</sub>)がエラーフレームであり、AFLM(f<sub>n+1</sub>)が有効データフレームエラーフレームであるときの例である。これも図8の例と同様に、エラーフレームであるAFLM(f<sub>n-1</sub>)内の振幅データEXP(f<sub>n+1</sub>, 5, band)151を0と解釈すれば、図8の例のように滑らかなフェードインをさせることができる。図10は図1の処理方法を実現する圧縮音声再生装置の一例で、音声ストリーム200を入力し、ストリーム中の同期ワードSYNC102を検出して音声フレーム101の同期処理を行い、音声フレーム202内の誤り検出ワードEDC110を利用して誤り検出をし、誤りがあるかないかを示すフレームエラーフラグ205を出力するフレーム同期検出及び回路誤り検出201と、同期処理誤り検出後の音声フレームデータ202からデコード処理のために必要な情報を抜き出して保持するフレーム内情報抜き出し回路203と、フレーム内情報抜き出し回路203から振幅値コード208を入力してデータ振幅値をデコードする振幅値デコード回路209と、振幅値デコード回路209から出力されるデコード

された振幅値210を入力し、上記フレーム内情報抜き出し回路203から正規化サンプル値204を入力して、逆量子化処理を行う逆量子化回路214と、逆量子化回路214から出力されるサンプル再構成値216を各帯域について入力し、帯域合成して、最終的な時間領域信号218に変換して出力する帯域合成回路217とで構成されており、更に本発明の特徴部分として、フレーム同期および誤り検出回路201からのフレームエラーフラグ205を入力して保持する誤り履歴保持回路206と、振幅値デコード回路209でデコードされた振幅値のうち、圧縮音声フレーム内での最終オーディオデータブロックに対応する振幅値224を入力して保持する過去フレーム最終ブロック保持回路220と、(f<sub>n</sub>)番目圧縮音声フレームに対する逆量子化を行うときに(f<sub>n</sub>は整数)、(f<sub>n-1</sub>)番目圧縮音声フレームの最終オーディオデータブロックに対応する振幅値211を過去フレーム最終ブロック保持回路220から入力し、また上記振幅値デコード手段から(f<sub>n+1</sub>)番目圧縮音声フレームの先頭オーディオデータブロックに対応する振幅値226を入力して、振幅値補償データ213を生成して出力する補償値生成回路211と、誤り履歴保持回路206から(f<sub>n</sub>)番目圧縮音声フレームに対するフレームエラーフラグ219を入力して、フレームエラーフラグ219がエラーでないことを示す場合は、上記振幅値デコード回路209が出力する(f<sub>n</sub>)番目圧縮音声フレームの各オーディオデータブロックに対応する振幅値210を、上記フレームエラーフラグがエラーであることを示す場合は補償値生成回路211が出力する補償値としての各オーディオデータブロックに対応する振幅値213を選び、逆量子化回路214の入力207として出力する。この装置における振幅値補償データの生成に関連したデータ変化タイミングの一例が図11に示してある。同期検出、誤り検出、情報抜き出し用フレームデータ152と、保持最終ブロックデータ155と、サンプル逆量子化、帯域合成用フレーム156とが示してあり、エラーフレームAFLM(f<sub>n</sub>)内の振幅値EXPをAFLM(f<sub>n-1</sub>)内の振幅データEXP(f<sub>n-1</sub>, 5, band)153とAFLM(f<sub>n+1</sub>)内の振幅データEXP(f<sub>n+1</sub>, 0, band)154から生成する場合のタイミングの一例である。このタイミングにより補償値が算出でき、AFLM(f<sub>n-1</sub>)内の振幅データEXP(f<sub>n-1</sub>, 5, band)153とAFLM(f<sub>n</sub>)内の補償後振幅データEXP(f<sub>n</sub>, 0, band)158、またAFLM(f<sub>n</sub>)内の補償後振幅データEXP(f<sub>n</sub>, 5, band)159とAFLM(f<sub>n-1</sub>)内の振幅データEXP(f<sub>n-1</sub>, 0, band)160が滑らかにつながる。この構成により、図1の例で示した補償方法を圧縮音声再生装置において実現する事ができる。このようにして、エラーフレームに対するエラー補償処理が行

われ、時間域での滑らかなつながりだけでなく、周波数域でのスペクトル分布の滑らかなつながりが実現できるため、聴感上の劣化を抑制できる。

【発明の効果】エラーの検出された圧縮音声フレームのエラー補償処理が、より細やかに行われ、フレーム全体が訂正不能な状態なのに対して、フレーム内のブロックごとのデータ補償を行うことができる。これにより、1フレームのみ(2フレーム以上連続しない)のエラーの場合には、前フレーム完全繰返しよりもエラー補償フレームからの復帰部分が周波数軸上で滑らかにつながる。また、1フレーム完全ミュートに比べて、エラー部分の音圧レベルの抑えすぎにならないので、データ補償処理発生時の聴感の劣化の抑制が期待できる。また、この処理のために保持すべきデータは1ブロック分の振幅係数で済むため、ハードウェアの増加も少ない。

#### 【図面の簡単な説明】

【図1】本発明を適用した圧縮音声エラー補償処理方法の一実施例を示すフローチャート。

【図2】本発明のエラー補償処理詳細部分の一例を示すフローチャート。

【図3】本発明を適用した圧縮音声エラー補償処理方法に対する入力としての圧縮音声ストリームの構造とデコード結果音声の一例を示す図。

【図4】本発明を適用した圧縮音声エラー補償処理方法がエラー補償処理を行う状況の圧縮音声フレームとデコ

ード結果音声の一例を示す図。

【図5】本発明を適用した圧縮音声エラー補償処理方法で生成される補償値の、隣接フレーム内のデータとの関連を示した一例を示す図。

【図6】従来の補償処理であるフレーム全体繰返しの一例を示す図。

【図7】従来の補償処理であるフレーム全体ミュートの一例を示す図。

【図8】本発明を適用した圧縮音声エラー補償処理方法で生成される補償値の別の一例で、有効フレームからエラーフレームが2つ以上連続する場合の例を示す図。

【図9】本発明を適用した圧縮音声エラー補償処理方法で生成される補償値の別の一例で、エラーフレームが2つ以上連続した後に有効フレームが来た場合の例を示す図。

【図10】本発明を適用した圧縮音声エラー補償処理装置の一実施例を示すフローチャート。

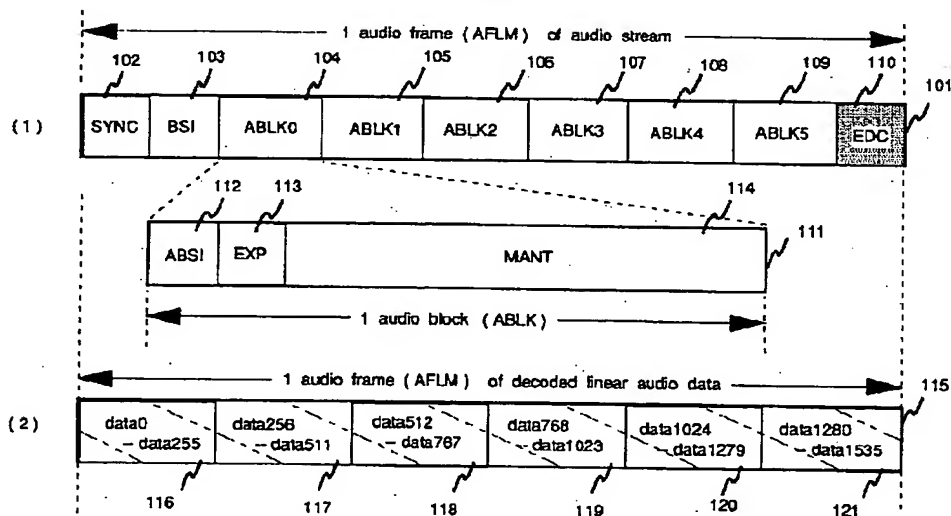
【図11】本発明を適用した圧縮音声エラー補償処理装置における補償値生成のためのタイミング。

#### 【符号の説明】

2…音声フレーム入力、3…フレームエラー検出、結果保存、4…逆量子化用フレームエラー有無確認、5…振幅値データ再現、6…サンプル逆量子化、7…最終ブロックデータ保存、10…(fn-1)番フレーム最終ブロック呼び戻し、11…振幅データ補償値の算出。

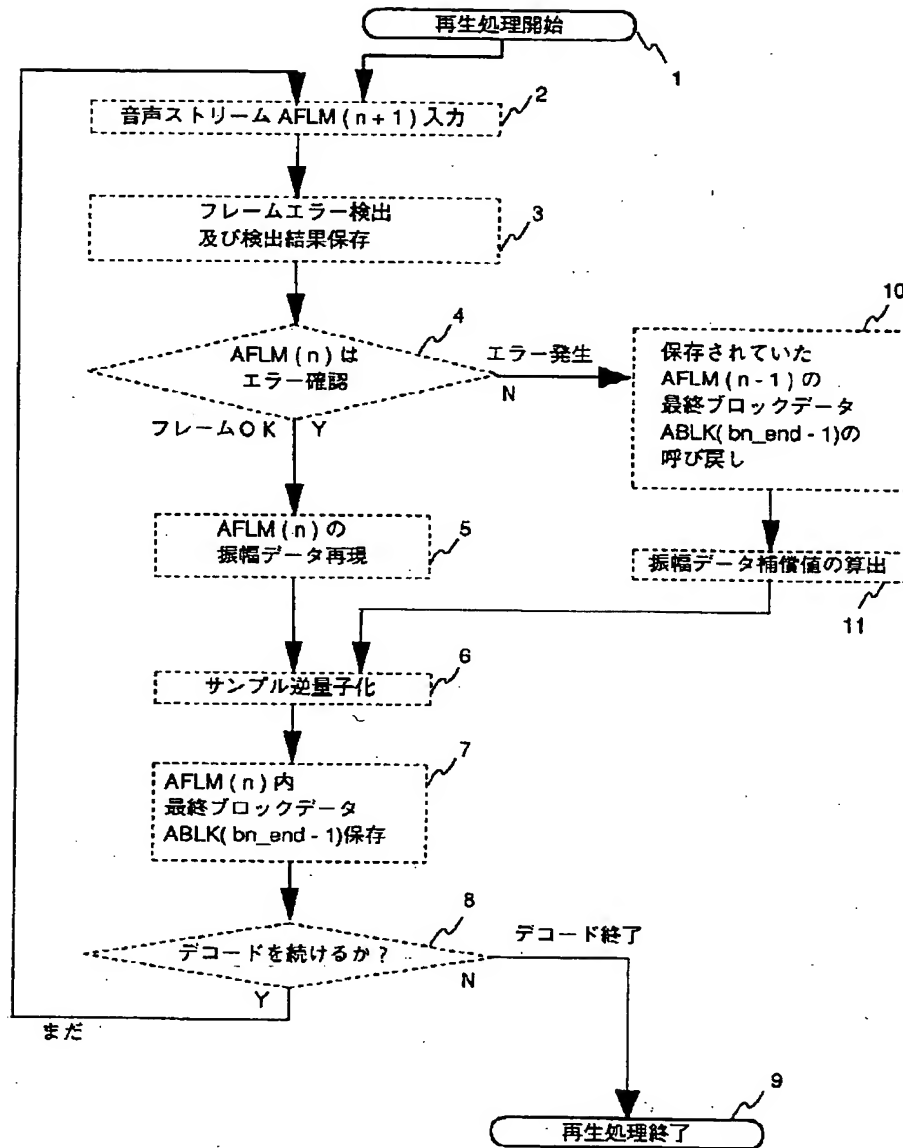
【図3】

図 3



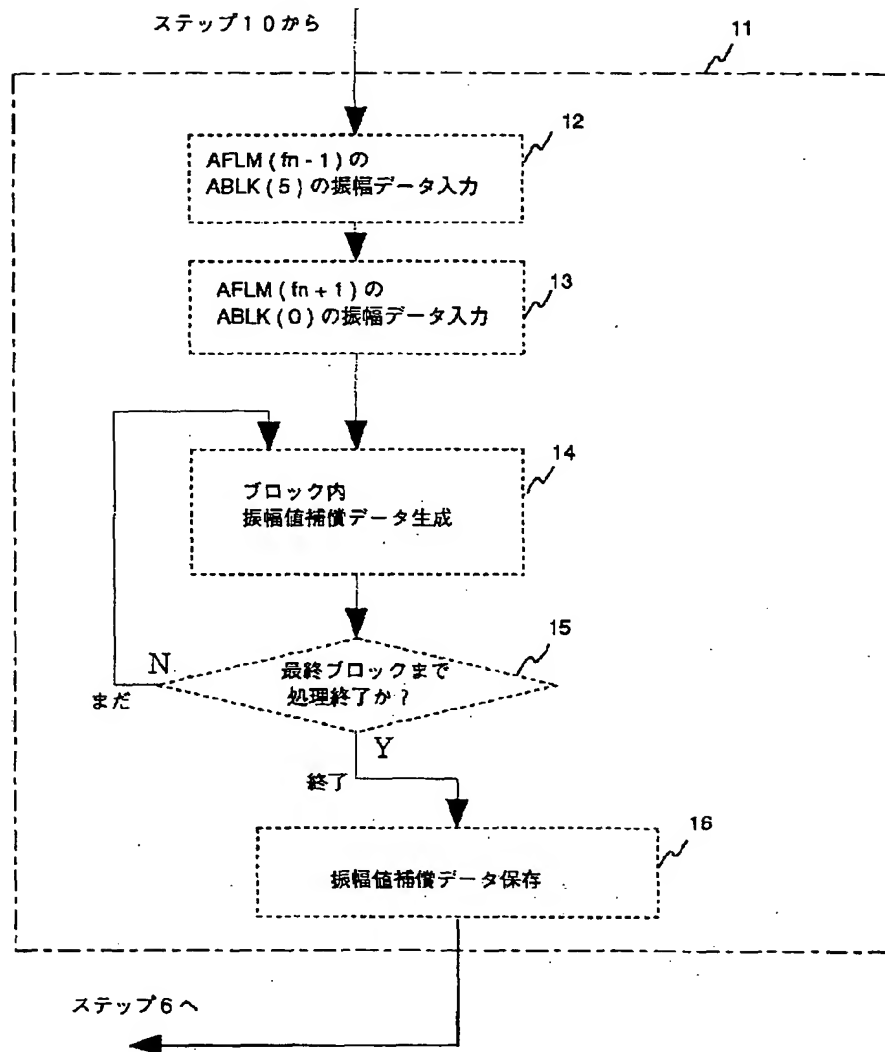
【図1】

図 1



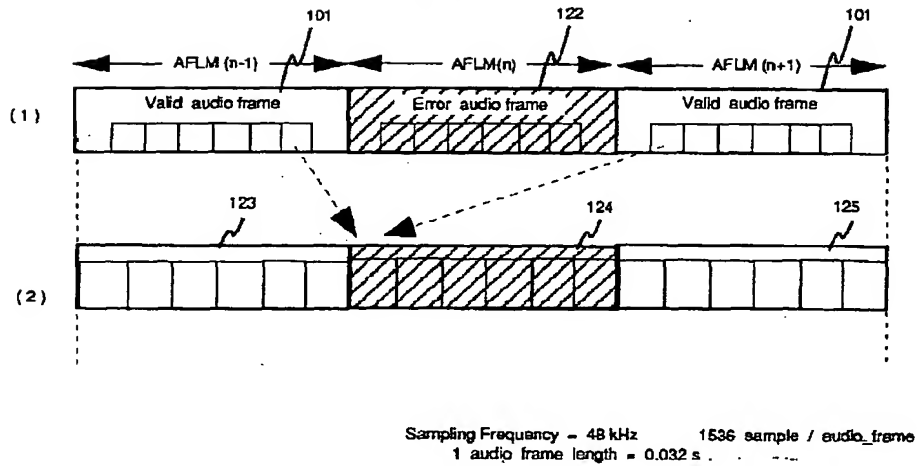
【図2】

図 2



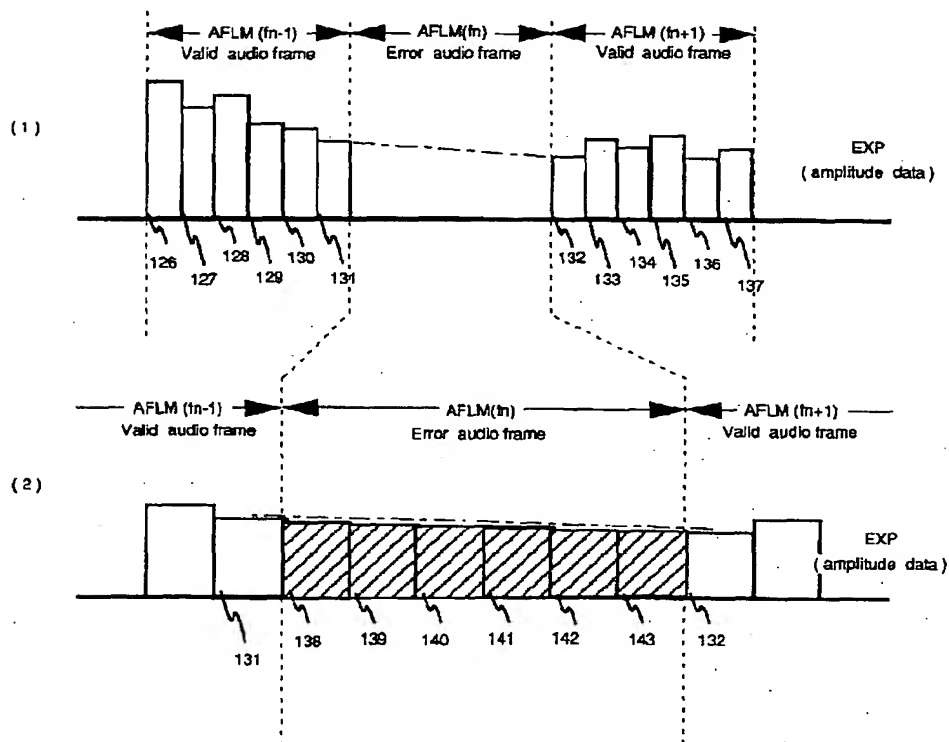
【図4】

図4



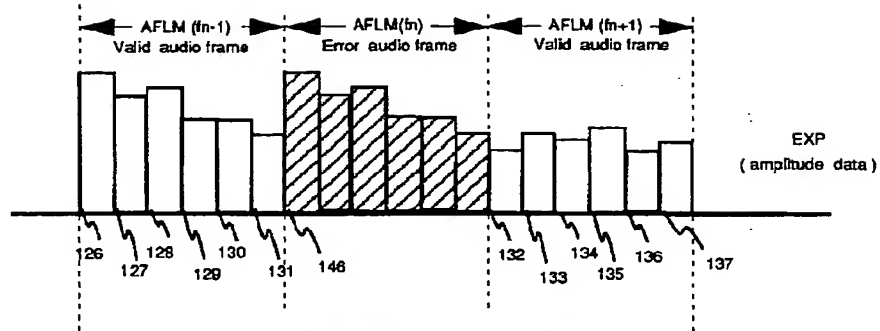
【図5】

図5



【図6】

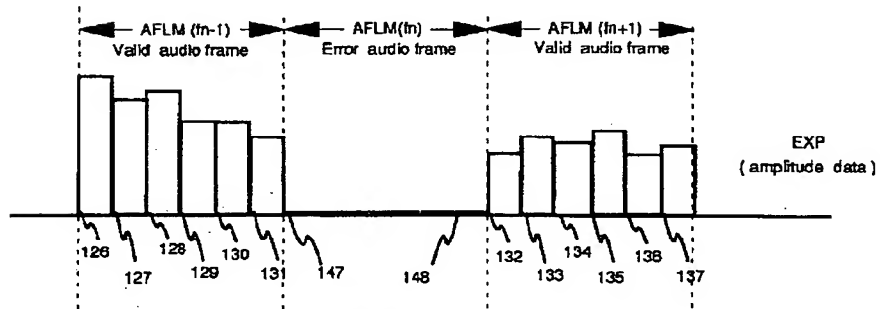
図 6



「従来データ補償処理その1： フレーム繰り返し」

【図7】

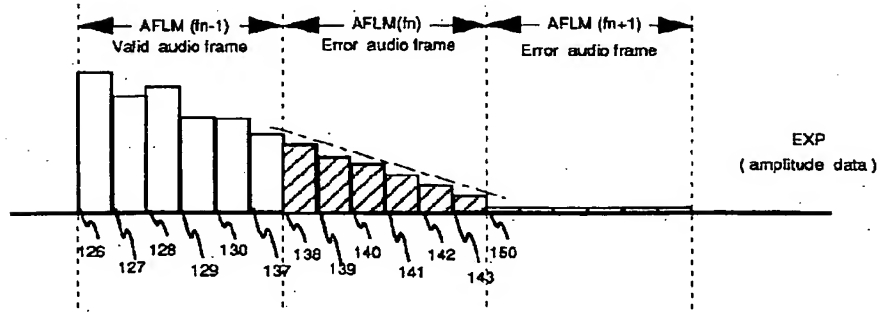
図 7



「従来データ補償処理その2： フレームミュート」

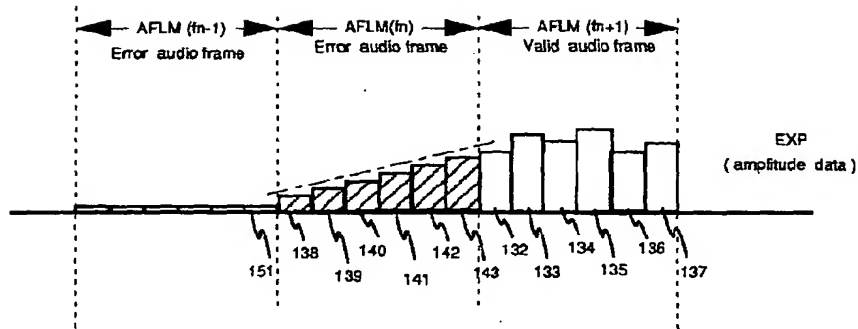
【図8】

図 8



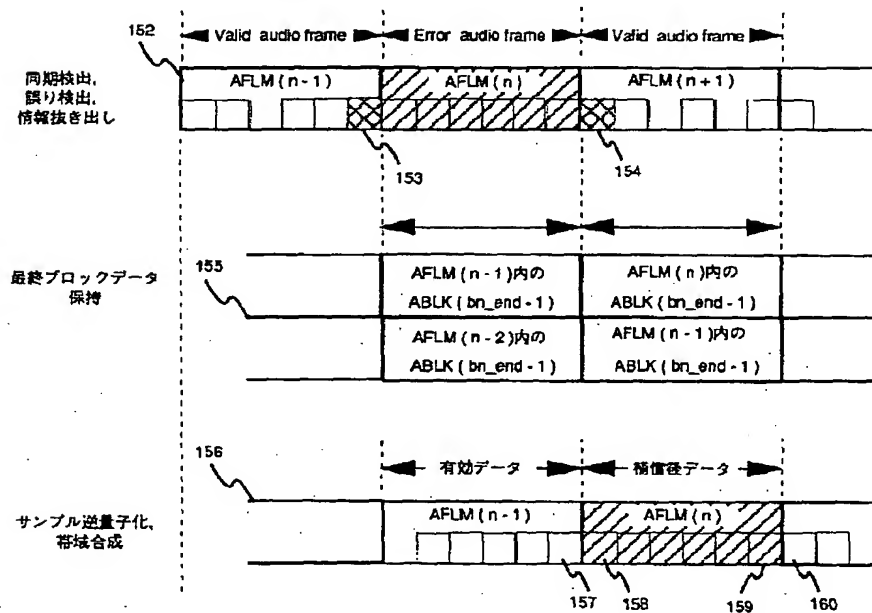
【図9】

図9



【図11】

図11



【図10】

図 10

